PEAR: a Tool for Reasoning About Scenarios and Probabilities in Description Logics of Typicality

Gian Luca Pozzato[1]* and Gabriele Soriano[1]

Dipartimento di Informatica, Università di Torino, Italy - gianluca.pozzato@unito.it,gabriele.soriano@edu.unito.it

Abstract. In this work we describe PEAR, a tool for reasoning about prototypical properties in an extension of Description Logics of typicality with probabilities and scenarios. PEAR implements a non-monotonic procedure for the logic $\mathcal{ALC}+\mathbf{T}_{\mathbf{R}}^{\mathsf{P}}$, a recently introduced extension of the logic of typicality $\mathcal{ALC}+\mathbf{T}_{\mathbf{R}}$ by inclusions of the form $\mathbf{T}(C) \sqsubseteq_p D$, where p is a real number between 0 and 1 capturing the intuition that "all the typical Cs are Ds, and the probability that a C is not a D is $1-p$". In this logic, different scenarios are considered by taking into account several extension of the ABox, containing only some typicality assumptions about individuals. Each scenario has a probability depending on those equipping typicality inclusions, then entailment can be restricted to scenarios whose probabilities belong to a given and fixed range. PEAR is implemented in Python, it computes all scenarios of a knowledge base and it allows the user to check the probability of a query by exploiting a translation into standard $\mathcal{ALC}$.

1 Introduction

Non-monotonic extensions of Description Logics (from now on, DLs for short) have been actively investigated since the early 90s [1–6], in order to tackle the well known problem of allowing one to represent and reason about prototypical properties and defeasible inheritance. A simple but powerful non-monotonic extension of DLs is proposed in [7]: in this approach, typical properties can be directly specified by means of a "typicality" operator: a TBox, in addition to standard inclusions $C \sqsubseteq D$, representing that "all Cs are also Ds", can contain inclusions of the form $\mathbf{T}(C) \sqsubseteq D$ to represent that "typical Cs are also Ds". The Description Logic so obtained is called $\mathcal{ALC}+\mathbf{T}_{\mathbf{R}}$ and, as a difference with standard DLs, one can consistently express exceptions and reason about defeasible inheritance as well. For instance, a knowledge base can consistently contain the inclusions (1) $\mathit{HummingBird} \sqsubseteq \mathit{Bird}$, (2) $\mathbf{T}(\mathit{Bird}) \sqsubseteq \neg \mathit{BackwardsFlier}$ and (3) $\mathbf{T}(\mathit{HummingBird}) \sqsubseteq \mathit{BackwardsFlier}$, expressing that, normally, birds are not able to fly backwards, however hummingbirds are exceptional birds that, typically, are able to fly backwards. The key point of the logic $\mathcal{ALC}+\mathbf{T}_{\mathbf{R}}$ relies

* The author is partially supported by the project "METALLIC #2: METodi di prova per il ragionamento Automatico per Logiche non-cLassICHe #2", INdAM GNCS 2019.

on the fact that the semantics of the $\mathbf{T}$ operator is strongly related to axioms and rules of *rational entailment* as introduced in [8], widely considered as the core properties of non-monotonic reasoning. As a consequence, $\mathbf{T}$ inherits well-established properties like *specificity*: in the example, if one knows that Petey is a typical hummingbird, then the logic $\mathcal{ALC} + \mathbf{T_R}$ allows one to infer that it flies backwards, giving preference to the most specific information of being a hummingbird with respect to being a bird.

The main drawback of the logic $\mathcal{ALC} + \mathbf{T_R}$ it that it is too weak: indeed, although the operator $\mathbf{T}$ is non-monotonic ($\mathbf{T}(C) \sqsubseteq E$ does not imply $\mathbf{T}(C \sqcap D) \sqsubseteq E$), the logic $\mathcal{ALC} + \mathbf{T_R}$ is monotonic [9, 10], namely, if F is entailed from a knowledge base KB, then F is also entailed from any KB' $\supseteq$ KB. As a consequence, unless a KB contains explicit assumptions about typicality of individuals, there is no way of inferring defeasible properties about them: in the above example, if KB only contains that Petey is a hummingbird ($\mathit{HummingBird}(\mathit{petey})$), it is not possible to conclude that it flies backwards: this would be possible only if the stronger information that Petey is a *typical* hummingbird ($\mathbf{T}(\mathit{HummingBird})(\mathit{petey})$) belongs to (or can be inferred from) KB. In order to tackle this problem, in [9, 11] the authors have strengthened the semantics of the logic $\mathcal{ALC}+\mathbf{T_R}$ by means of a minimal model semantics, corresponding to a notion of *rational closure* as defined in [8] for propositional logic. Intuitively, the idea is to restrict reasoning to (canonical) models that maximize typical instances of a concept when consistent with the knowledge base. As a consequence, in the resulting logic $\mathcal{ALC} + \mathbf{T}_{\mathbf{R}}^{RaCl}$, if one knows that Petey is a hummingbird, one can nonmonotonically assume that it is also a *typical* one if this is consistent, and therefore that it is able to fly backwards. From a semantic point of view, the logic $\mathcal{ALC} + \mathbf{T}_{\mathbf{R}}^{RaCl}$ is based on a preference relation among $\mathcal{ALC}+\mathbf{T_R}$ models and a notion of *minimal entailment* restricted to models that are minimal with respect to such preference relation.

The logic $\mathcal{ALC} + \mathbf{T}_{\mathbf{R}}^{RaCl}$ imposes to consider *all* typicality assumptions that are consistent with a KB. In other words, in absence of explicit contradictory information, the logic $\mathcal{ALC}+\mathbf{T}_{\mathbf{R}}^{RaCl}$ leads to assume that there are no exceptions: this seems to be too strong in several application domains, especially in situations characterized by a high number of individuals that are all considered as typical ones, whereas it could be useful to reason about scenarios exhibiting *exceptional ones*. To this aim, in [12] we have introduced a further extension of the logic of typicality called $\mathcal{ALC}+\mathbf{T}_{\mathbf{R}}^{\mathsf{P}}$, which extends $\mathcal{ALC}$ by means of typicality inclusions equipped by *probabilities* of the form $\mathbf{T}(C) \sqsubseteq_p D$, where $p \in (0, 1)$, whose intuitive meaning is that "normally, Cs are Ds and the probability of having exceptional Cs – not being Ds – is $1 - p$". All the typical instances of the concept C are also instances of the concept D, but we have the opportunity of expressing a degree about the fact that a C element is not also a D element, i.e. it is an exceptional C element. This allows us to rank different typicality inclusions, for instance we can have that $\mathbf{T}(\mathit{TeenAger}) \sqsubseteq_{0.6} \mathit{InfluencerFollower}$ as well as that $\mathbf{T}(\mathit{TeenAger}) \sqsubseteq_{0.9} \exists \mathit{play}.\mathit{VideoGame}$, capturing the intuition that both following an influencer and playing video games are typical properties of teenagers, and we also want to express that the probability of having exceptional teenagers not following any influencer is higher than the one of finding ones not playing video games, by ranking the two properties with probabilities 0.6 and 0.9, respectively.

In this work we introduce PEAR, a preliminary implementation of the reasoning mechanism for the logic $\mathcal{ALC} + \mathbf{T}_{\mathbf{R}}^{\mathsf{P}}$ described in [12]. PEAR, standing for Probability of Exceptions and typicAlity Reasoner, computes different syntactic extensions of an ABox containing only some of the "plausible" typicality assertions that can be entailed from the KB in the non-monotonic logic $\mathcal{ALC} + \mathbf{T}_{\mathbf{R}}^{RaCl}$: each extension represents a *scenario* having a specific probability. Then, PEAR allows one to check whether a query is non-monotonically entailed from the knowledge base, either by selecting the scenarios where it holds or by restricting the attention to extensions whose probabilities belong to a given and fixed range, as well as to compute the probability of a query as the sum of the probabilities of scenarios from which it is entailed. This allows one to reason about scenarios that are not necessarily the most probable and are, in some sense, *surprising*. The decision procedure computed by PEAR is ExpTime complete, therefore we retain the same complexity of the underlying standard $\mathcal{ALC}$.

2 Preferential Description Logics

Let us first quickly recall the main notions about the Description Logic of typicality $\mathcal{ALC} + \mathbf{T}_{\mathbf{R}}$ introduced in [7, 13, 9]. The logic $\mathcal{ALC} + \mathbf{T}_{\mathbf{R}}$ is obtained by adding to standard $\mathcal{ALC}$ the typicality operator $\mathbf{T}$ [7]. The intuitive idea is that $\mathbf{T}(C)$ selects the *typical* instances of a concept C. We can therefore distinguish between the properties that hold for all instances of concept C ($C \sqsubseteq D$), and those that only hold for the normal or typical instances of C ($\mathbf{T}(C) \sqsubseteq D$). From a semantic point of view, we refer to *rational models* [9]: a model $\mathcal{M}$ is any structure $\langle \Delta^{\mathcal{I}}, <, .^{\mathcal{I}} \rangle$ where $\Delta^{\mathcal{I}}$ is the domain, $<$ is an irreflexive, transitive, well-founded and modular (for all x, y, z in $\Delta^{\mathcal{I}}$, if $x < y$ then either $x < z$ or $z < y$) relation over $\Delta^{\mathcal{I}}$. In this respect, $x < y$ means that x is "more normal" than y, and that the typical members of a concept C are the minimal elements of C with respect to this relation. An element $x \in \Delta^{\mathcal{I}}$ is a *typical instance* of some concept C if $x \in C^{\mathcal{I}}$ and there is no C-element in $\Delta^{\mathcal{I}}$ *more typical* than x. In detail, $.^{\mathcal{I}}$ is the extension function that maps each concept C to $C^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$, and each role R to $R^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$. For concepts of $\mathcal{ALC}$, $C^{\mathcal{I}}$ is defined as usual. For the $\mathbf{T}$ operator, let $Min_<(C^{\mathcal{I}}) = \{x \in C^{\mathcal{I}} \mid \nexists y \in C^{\mathcal{I}} \text{ s.t. } y < x\}$, we define $(\mathbf{T}(C))^{\mathcal{I}} = Min_<(C^{\mathcal{I}})$.

A model $\mathcal{M}$ can be equivalently defined by postulating the existence of a function $k_{\mathcal{M}} : \Delta^{\mathcal{I}} \longmapsto \mathbb{N}$, where $k_{\mathcal{M}}$ assigns a finite rank to each domain element: the rank function $k_{\mathcal{M}}$ and $<$ can be defined from each other by stating that $x < y$ if and only if $k_{\mathcal{M}}(x) < k_{\mathcal{M}}(y)$. Given standard definitions of satisfiability of a KB in a model, we define a notion of entailment in $\mathcal{ALC} + \mathbf{T}_{\mathbf{R}}$. Given a query F (either an inclusion $C \sqsubseteq D$ or an assertion $C(a)$ or an assertion of the form $R(a, b)$), we say that F is entailed from a KB if F holds in all $\mathcal{ALC} + \mathbf{T}_{\mathbf{R}}$ models satisfying KB.

As already mentioned in the Introduction, even if the $\mathbf{T}$ operator itself is non-monotonic, the logic $\mathcal{ALC} + \mathbf{T_R}$ is monotonic. In order to perform useful non-monotonic inferences, in [9] the authors have strengthened the above semantics by restricting entailment to a class of minimal models. Intuitively, the idea is to restrict entailment to models that *minimize the untypical instances of a concept.* The resulting logic is called $\mathcal{ALC} + \mathbf{T}_{\mathbf{R}}^{RaCl}$ and it corresponds to a notion of *rational closure* on top of $\mathcal{ALC} + \mathbf{T_R}$. Such a notion is a natural extension of the rational closure construction provided in [8] for the propositional logic. The non-monotonic semantics of $\mathcal{ALC}+\mathbf{T}_{\mathbf{R}}^{RaCl}$ relies on minimal rational models that minimize the *rank of domain elements*. Informally, given two models of KB, one in which a given domain element x has rank 2 (because for instance $z < y < x$), and another in which it has rank 1 (because only $y < x$), we prefer the latter, as in this model the element x is assumed to be "more typical" than in the former. Query entailment is then restricted to minimal *canonical models*. The intuition is that a canonical model contains all the individuals that enjoy properties that are consistent with KB. A model $\mathcal{M}$ is a minimal canonical model of KB if it satisfies KB, it is minimal and it is canonical[1]. A query F is minimally entailed from a KB if it holds in all minimal canonical models of KB. In order to ascribe typical properties to individuals, the notion of rational closure is further extended to the ABox. In [9] it is shown that query entailment in $\mathcal{ALC}+\mathbf{T}_{\mathbf{R}}^{RaCl}$ is in ExpTime.

3 The Logic $\mathcal{ALC} + \mathbf{T}_{\mathbf{R}}^{\mathsf{P}}$

In this section we recall the logic $\mathcal{ALC} + \mathbf{T}_{\mathbf{R}}^{\mathsf{P}}$, extending the DLs of typicality with probabilities and scenarios. Here, typicality inclusion are equipped with a probability/degree of *not* having exceptions for it, and have the form

$$\mathbf{T}(C) \sqsubseteq_p D,$$

whose meaning is "typical Cs are also Ds, and the probability of having exceptional Cs not being Ds is $1 - p$".

Definition 1. *We consider an alphabet of concept names $\mathcal{C}$, of role names $\mathcal{R}$, and of individual constants $\mathcal{O}$. Given $A \in \mathcal{C}$ and $R \in \mathcal{R}$, we define:*

$$C := A \mid \top \mid \bot \mid \neg C \mid C \sqcap C \mid C \sqcup C \mid \forall R.C \mid \exists R.C$$

An $\mathcal{ALC} + \mathbf{T}_{\mathbf{R}}^{\mathsf{P}}$ knowledge base is a pair $(\mathcal{T}, \mathcal{A})$. $\mathcal{T}$ contains axioms of the form either $C \sqsubseteq C$ or $\mathbf{T}(C) \sqsubseteq_p C$, where $p \in \mathbb{R}, p \in (0, 1)$. $\mathcal{A}$ contains assertions of the form either $C(a)$ or $R(a, b)$, where $a, b \in \mathcal{O}$.

Given an inclusion $\mathbf{T}(C) \sqsubseteq_p D$, the higher the probability p, the less the probability of having exceptional Cs not being also Ds. In this respect, the probability p is a real number included in the open interval $(0, 1)$: the probability 1 is not allowed, in the sense that an inclusion $\mathbf{T}(C) \sqsubseteq_1 D$ would correspond to a *strict*

[1] In Theorem 10 in [9] the authors have shown that for any consistent KB there exists a finite minimal canonical model of KB.

inclusion $C \sqsubseteq D$ (all Cs are Ds). Given another inclusion $\mathbf{T}(C') \sqsubseteq_{p'} D'$, with $p' < p$, we assume that this inclusion is less "strict" than the other one, i.e. the probability of having exceptional C's is higher than the one of having exceptional Cs with respect to properties D' and D, respectively.

Given a KB, we define the finite set $\mathfrak{Tip}$ of concepts occurring in the scope of the typicality operator, i.e. $\mathfrak{Tip} = \{C \mid \mathbf{T}(C) \sqsubseteq_p D \in \text{ KB}\}$. Given an individual a explicitly named in the ABox, we define the set of typicality assumptions $\mathbf{T}(C)(a)$ that can be minimally entailed from KB in the non-monotonic logic $\mathcal{ALC} + \mathbf{T}_{\mathbf{R}}^{RaCl}$, with $C \in \mathfrak{Tip}$. We then consider an ordered set $\mathfrak{Tip}_{\mathcal{A}}$ of pairs (a, C) of all possible assumptions $\mathbf{T}(C)(a)$, for all concepts $C \in \mathfrak{Tip}$ and all individual constants a in the ABox. We then define the *ordered multiset* $\mathcal{P}_{\mathcal{A}} = [p_1, p_2, \ldots, p_n]$, respecting the order imposed on $\mathfrak{Tip}_{\mathcal{A}}$, where $p_i = \prod_{j=1}^{m} p_{ij}$ for all $\mathbf{T}(C_i) \sqsubseteq_{p_{i1}} D_1, \mathbf{T}(C_i) \sqsubseteq_{p_{i2}} D_2, \ldots, \mathbf{T}(C_i) \sqsubseteq_{p_{im}} D_m$ in $\mathcal{T}$. The ordered multiset $\mathcal{P}_{\mathcal{A}}$ is a tuple of the form $[p_1, p_2, \ldots, p_n]$, where p_i is the probability of the assumption $\mathbf{T}(C)(a)$, such that $(a, C) \in \mathfrak{Tip}_{\mathcal{A}}$ at position i. p_i is the product of all the probabilities p_{ij} of inclusions $\mathbf{T}(C) \sqsubseteq_{p_{ij}} D$ in the TBox.

Following the basic idea underlying surprising scenarios outlined in [14], we consider different extensions $\widetilde{\mathcal{A}_i}$ of the ABox and we equip them with a probability $\mathbb{P}_i$. Starting from $\mathcal{P}_{\mathcal{A}} = [p_1, p_2, \ldots, p_n]$, the first step is to build all alternative tuples where 0 is used in place of some p_i to represent that the corresponding typicality assertion $\mathbf{T}(C)(a)$ is no longer assumed. We then define the notion of *extension* of the ABox corresponding to a string so obtained: in this way, the extension corresponding to $\mathcal{P}_{\mathcal{A}}$ contain all the available typicality assumptions; in the other extensions, some typicality assumptions are discarded, thus 0 is used in place of the corresponding p_i. The probability of an extension $\widetilde{\mathcal{A}_i}$ corresponding to a string $\mathcal{P}_{\mathcal{A}_i} = [p_{i1}, p_{i2}, \ldots, p_{in}]$ is defined as the product of probabilities p_{ij} when $p_{ij} \neq 0$, i.e. the probability of the corresponding typicality assumption when this is selected for the extension, and $1 - p_j$ when $p_{ij} = 0$, i.e. the corresponding typicality assumption is discarded, that is to say the extension contains an exception to the inclusion. It is easy to observe that we obtain a probability distribution over extensions of $\mathcal{A}$.

Non-monotonic entailment of a query F in the Description Logic $\mathcal{ALC} + \mathbf{T}_{\mathbf{R}}^{\mathsf{P}}$ is defined as follows: (i) if F is a TBox inclusion $C \sqsubseteq D$, then it is entailed from KB if it is minimally entailed from KB$'$ in the non-monotonic $\mathcal{ALC} + \mathbf{T}_{\mathbf{R}}^{RaCl}$, where KB$'$ is obtained from KB by removing probabilities of exceptions, i.e. by replacing each typicality inclusion $\mathbf{T}(C) \sqsubseteq_p D$ with $\mathbf{T}(C) \sqsubseteq D$; (ii) if F is an ABox fact $C(a)$, then it is entailed from KB if it is entailed in the monotonic $\mathcal{ALC} + \mathbf{T}_{\mathbf{R}}$ from scenarios extending KB with $\mathcal{A}_i$. Furthermore, one can consider a notion of entailment restricted to scenarios whose probabilities belong to a given range and, similarly to [15], a notion of probability of the entailment of a query $C(a)$, as the sum of the probabilities of all extensions from which $C(a)$ is so entailed. In [12] it is shown that entailment in $\mathcal{ALC} + \mathbf{T}_{\mathbf{R}}^{\mathsf{P}}$ is ExpTime-complete as for standard $\mathcal{ALC}$.

4 Design of PEAR

In this section we introduce PEAR (Probability of Exceptions and typicAlity Reasoner), a Python implementation of the reasoning services provided by the logic $\mathcal{ALC} + \mathbf{T}^{\mathsf{P}}_{\mathbf{R}}$. PEAR makes use of the library `owlready2` [2] that allows one to rely on the services of efficient DL reasoners, e.g. the HermiT reasoner. PEAR exploits the translation of an $\mathcal{ALC} + \mathbf{T_R}$ knowledge base into standard $\mathcal{ALC}$ introduced in [9] and described later in this section. The current version of the system PEAR, along with the files for the examples presented in this paper, are available at http://di.unito.it/pear.

The system PEAR implements a non-monotonic procedure whose aim is to compute extensions $\mathcal{A}$ of the ABox and corresponding alternative scenarios having different probabilities described in the previous section. Given an $\mathcal{ALC} + \mathbf{T}^{\mathsf{P}}_{\mathbf{R}}$ KB=$(\mathcal{T}, \mathcal{A})$ and a query F, PEAR computes the following four steps: 1. compute the set $\mathfrak{Tip}_a$ of all typicality assumptions that are minimally entailed from the knowledge base in the non-monotonic logic $\mathcal{ALC}+\mathbf{T}^{RaCl}_{\mathbf{R}}$; 2. compute all possible $\widetilde{A_i}$ extensions of the ABox and compute their probabilities; 3. select the extensions whose probabilities belong to a given range $\langle p, q\rangle$; 4. check whether the query F is entailed from all the selected extensions in the monotonic logic $\mathcal{ALC}+\mathbf{T_R}$. Step 1 is based on reasoning in the non-monotonic logic $\mathcal{ALC}+\mathbf{T}^{RaCl}_{\mathbf{R}}$: in this case, PEAR computes the rational closure of an $\mathcal{ALC} + \mathbf{T_R}$ knowledge base by means of the algorithm introduced in [9], which is sound and complete with respect to the minimal model semantics recalled in Section 2. Step 4 is based on reasoning in the monotonic logic $\mathcal{ALC}+\mathbf{T_R}$: to this aim, the procedure relies on a polynomial encoding of $\mathcal{ALC}+\mathbf{T_R}$ into $\mathcal{ALC}$ introduced in [16], which exploits the definition of $\mathbf{T}$ in terms of a Gödel-Löb modality $\Box$ as follows: $\mathbf{T}(C)$ is defined as $C \sqcap \Box\neg C$ where the accessibility relation of the modality $\Box$ is the preference relation $<$ in $\mathcal{ALC}+\mathbf{T_R}$ models. Let KB=$(\mathcal{T}, \mathcal{A})$ be a knowledge base where $\mathcal{A}$ does not contain positive typicality assertions on individuals of the form $\mathbf{T}(C)(a)$. The encoding KB'=$(\mathcal{T}', \mathcal{A}')$ of KB in $\mathcal{ALC}$ is defined as follows. First of all, we let $\mathcal{A}' = \emptyset$. Then, for each $A \sqsubseteq B \in \mathcal{T}$, not containing $\mathbf{T}$, we introduce $A \sqsubseteq B$ in $\mathcal{T}'$. For each $\mathbf{T}(A)$ occurring in $\mathcal{T}$, we introduce a new atomic concept $Box_{\neg A}$ and, for each inclusion $\mathbf{T}(A) \sqsubseteq_p B \in \mathcal{T}$, we add to $\mathcal{T}'$ the inclusion $A \sqcap Box_{\neg A} \sqsubseteq B$. In order to capture the properties of the $\Box$ modality, a new role R is introduced to represent the relation $<$ in preferential models, and the following inclusions are introduced in $\mathcal{T}'$: (i) $Box_{\neg A} \sqsubseteq \forall R.(\neg A \sqcap Box_{\neg A})$ and (ii) $\neg Box_{\neg A} \sqsubseteq \exists R.(A \sqcap Box_{\neg A})$. Inclusion (i) accounts for the transitivity of $<$, whereas inclusion (ii) accounts for the well-foundedness, namely the fact that if an element is not a typical A element then there must be a typical A element preferred to it. For the encoding of the inclusions, if $C_l \sqsubseteq C_r$ is not a typicality inclusion, then $C'_l = C_l$ and $C'_r = C_r$; if $C_l \sqsubseteq C_r$ is a typicality inclusion $\mathbf{T}(A) \sqsubseteq C_r$, then $C'_l = A \sqcap Box_{\neg A}$ and $C'_r = C_r$. The size of KB' is polynomial in the size of the KB. The same for C'_l and C'_r, assuming the size of C_l and C_r be polynomial in the size of KB. As an example, let the TBox contain:

[2] https://pythonhosted.org/Owlready2/

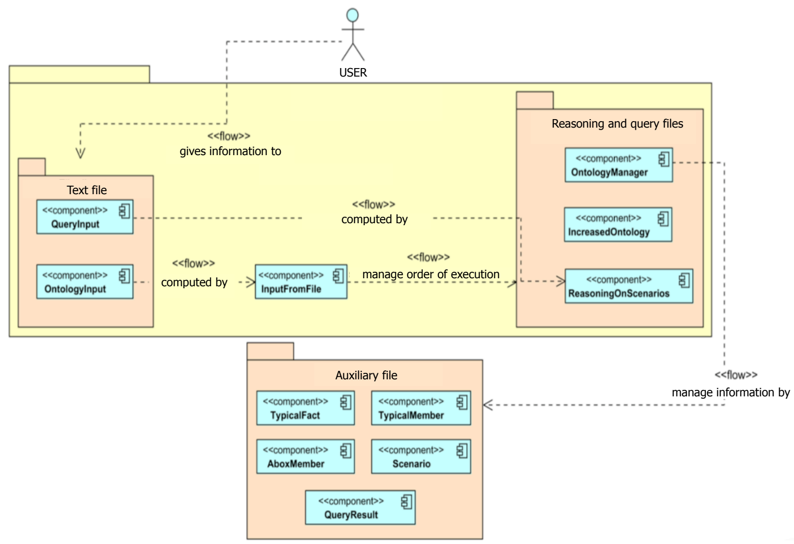

Fig. 1. The main components of the tool PEAR.

$$\begin{aligned}
&\mathit{HummingBird} \sqsubseteq \mathit{Bird}\\
&\mathbf{T}(\mathit{Bird}) \sqsubseteq \neg \mathit{BackwardsFlier}\\
&\mathbf{T}(\mathit{HummingBird}) \sqsubseteq \mathit{BackwardsFlier}
\end{aligned}$$

The system PEAR builds KB' containing:

$$\begin{aligned}
&\mathit{HummingBird} \sqsubseteq \mathit{Bird}\\
&\mathit{Bird} \sqcap \mathit{Bird1} \sqsubseteq \neg \mathit{BackwardsFlier}\\
&\mathit{Bird1} \sqsubseteq \forall R_1.(\neg \mathit{Bird} \sqcap \mathit{Bird1})\\
&\neg \mathit{Bird1} \sqsubseteq \exists R_1.(\mathit{Bird} \sqcap \mathit{Bird1})\\
&\mathit{HummingBird} \sqcap \mathit{HummingBird1} \sqsubseteq \mathit{BackwardsFlier}\\
&\mathit{HummingBird1} \sqsubseteq \forall R_2.(\neg \mathit{HummingBird} \sqcap \mathit{HummingBird1})\\
&\neg \mathit{HummingBird1} \sqsubseteq \exists R_2.(\mathit{HummingBird} \sqcap \mathit{HummingBird1})
\end{aligned}$$

The system PEAR comprises eleven Python files, whose behaviour and dependencies are summarized in Figure 4. In particular, the following modules represent the core components of the tool:

- `OntologyManager.py`, whose aim is to manage the knowledge base implementing the above translation into standard $\mathcal{ALC}$;
- `IncreasedOntology.py`, whose objectives are to generate all scenarios and to compute their probabilities;
- `ReasoningOnScenarios.py`, which is devoted to check entailment of a query (`QueryInput.py`) by exploiting reasoning services provided by HermiT.

Let us explain the functioning of PEAR by an example inspired by [12]. Let a KB be a knowledge base in $\mathcal{ALC} + \mathbf{T}_{\mathbf{R}}^{\mathsf{P}}$, where the TBox is as follows:

$$
\begin{aligned}
&\mathit{PokemonCardPlayer} \sqsubseteq \mathit{CardPlayer} && (1)\\
&\mathbf{T}(\mathit{CardPlayer}) \sqsubseteq_{0.85} \neg\, \mathit{YoungPerson} && (2)\\
&\mathbf{T}(\mathit{PokemonCardPlayer}) \sqsubseteq_{0.7} \mathit{YoungPerson} && (3)\\
&\mathbf{T}(\mathit{Student}) \sqsubseteq_{0.6} \mathit{YoungPerson} && (4)\\
&\mathbf{T}(\mathit{Student}) \sqsubseteq_{0.8} \mathit{InstagramUser} && (5)
\end{aligned}
$$

whereas the ABox is $\{\mathit{PokemonCardPlayer}(\mathit{lollo}), \mathit{Student}(\mathit{thomas})\}$. PEAR first computes the set of typicality assumptions entailed from $\mathcal{ALC} + \mathbf{T}_{\mathbf{R}}^{RaCl}$, namely (a) $\mathbf{T}(\mathit{PokemonCardPlayer})(\mathit{lollo})$ and (b) $\mathbf{T}(\mathit{Student})(\mathit{thomas})$. Then, PEAR generates the following four different scenarios:

1. neither (a) nor (b) are assumed: the probability is $(1-0.7)\times(1-(0.6\times0.8)) = 0.156$, but PEAR is not able to conclude anything about Lollo and Thomas;
2. (a) is assumed, whereas (b) is not: this scenario, having a probability $0.7 \times (1 - (0.6 \times 0.8)) = 0.364$, allows PEAR to conclude $\mathit{YoungPerson}(\mathit{lollo})$;
3. (b) is assumed, whereas (a) is not: this scenario has probability $(1 - 0.7) \times (0.6 \times 0.8) = 0.144$, and PEAR only concludes $\mathit{YoungPerson}(\mathit{thomas})$ and $\mathit{InstagramUser}(\mathit{thomas})$;
4. both (a) and (b) are assumed: in this scenario, whose probability is $0.7 \times (0.6 \times 0.8) = 0.336$. In this scenario, PEAR concludes that both Lollo and Thomas are young persons and that Thomas is an Instagram user.

PEAR can also evaluates the probability of a query as the sum of the probabilities of scenarios where such a query holds. In the example, PEAR evaluates the probability that Lollo is a young person as the sum of the probabilities of scenarios (2) and (4), and it is therefore $0.336 + 0.364 = 0.7$. Similarly, the probability that Thomas is an Instagram user is $0.336 + 0.144 = 0.48$. Some pictures of the execution of PEAR in this example are provided in Figure 4.

5 Related Works and Conclusions

In this work we have introduced PEAR, a tool for reasoning in the non-monotonic Description Logic $\mathcal{ALC} + \mathbf{T}_{\mathbf{R}}^{\mathsf{P}}$, which extends the logic of typicality by means of probabilities equipping typicality inclusions. In this setting, $\mathbf{T}(C) \sqsubseteq_p D$ is intended as "normally, Cs are Ds and we have a probability of $1 - p$ of having exceptional Cs not being Ds". From a knowledge representation point of view, as a difference from $\mathcal{ALC} + \mathbf{T}_{\mathbf{R}}^{RaCl}$, the logic $\mathcal{ALC} + \mathbf{T}_{\mathbf{R}}^{\mathsf{P}}$ allows one to distinguish among typicality inclusions by means of their probabilities: given two typical properties of a given concept, one can formalize the fact that the probability of having exceptional elements is different for the two properties/inclusions. Probabilities of exceptions are then used in order to reason about plausible scenarios, obtained by selecting only some typicality assumptions and whose probabilities belong to a given and fixed range. PEAR is implemented in Python and it provides a decision procedure for reasoning in the Description Logic $\mathcal{ALC} + \mathbf{T}_{\mathbf{R}}^{\mathsf{P}}$, which turns out to be in ExpTime as in the underlying standard $\mathcal{ALC}$.

An extension of DLs of typicality with probabilities is presented in [17], however probabilities are intended as degrees of belief as in the DISPONTE semantics [15].

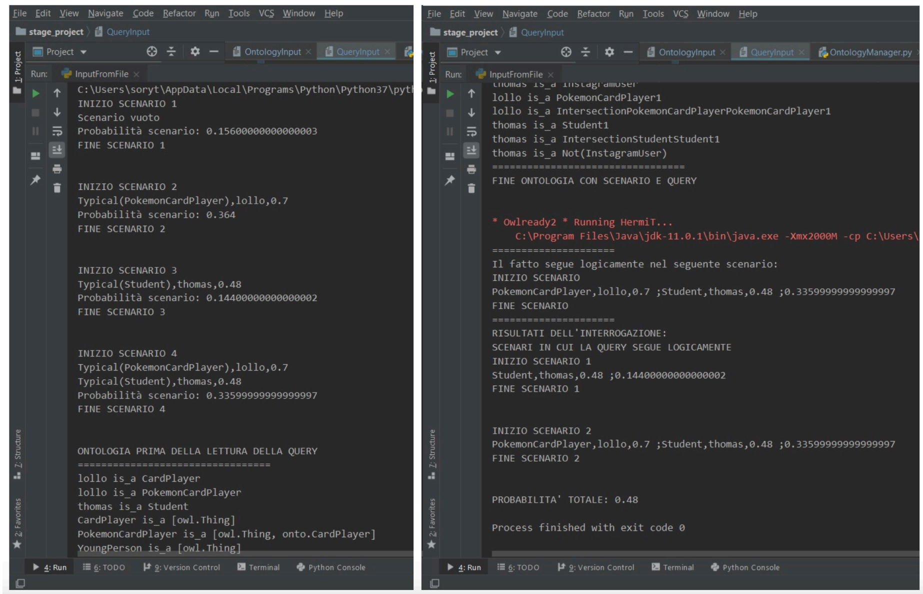

Fig. 2. Some pictures of PEAR. On the left, it can be observed the list of generated scenarios. On the right, PEAR shows the probability of the query $\mathit{InstagramUser}(\mathit{thomas})$.

PEAR represents a preliminary attempt to implement reasoning services for the logic $\mathcal{ALC} + \mathbf{T}^{\mathsf{P}}_{\mathbf{R}}$. We plan to provide a more mature version, by investigating the applicability of techniques introduced in [18, 19] in order to improve the efficiency of the system.

References

1. Bonatti, P.A., Lutz, C., Wolter, F.: The Complexity of Circumscription in DLs. Journal of Artificial Intelligence Research (JAIR) **35** (2009) 717–773
2. Baader, F., Hollunder, B.: Priorities on defaults with prerequisites, and their application in treating specificity in terminological default logic. JAR **15**(1) (1995) 41–68
3. Donini, F.M., Nardi, D., Rosati, R.: Description logics of minimal knowledge and negation as failure. ACM Transactions on Computational Logics (ToCL) **3**(2) (2002) 177–225
4. Casini, G., Straccia, U.: Rational closure for defeasible description logics. In: Proc. of JELIA 2010. Volume 6341 of LNAI., Springer (2010) 77–90
5. Casini, G., Straccia, U.: Defeasible Inheritance-Based Description Logics. Journal of Artificial Intelligence Research (JAIR) **48** (2013) 415–473
6. Bonatti, P.A., Faella, M., Petrova, I., Sauro, L.: A new semantics for overriding in description logics. Artificial Intelligence **222** (2015) 1–48
7. Giordano, L., Gliozzi, V., Olivetti, N., Pozzato, G.L.: ALC+T: a preferential extension of description logics. Fundamenta Informaticae **96** (2009) 341–372
8. Lehmann, D., Magidor, M.: What does a conditional knowledge base entail? Artificial Intelligence **55**(1) (1992) 1–60

9. Giordano, L., Gliozzi, V., Olivetti, N., Pozzato, G.L.: Semantic characterization of Rational Closure: from Propositional Logic to Description Logics. Artif. Intelligence **226** (2015) 1–33
10. Giordano, L., Gliozzi, V., Olivetti, N., Pozzato, G.L.: Minimal model semantics and rational closure in description logics. In Eiter, T., Glimm, B., Kazakov, Y., Krötzsch, M., eds.: Informal Proceedings of the 26th International Workshop on Description Logics. Volume 1014 of CEUR Workshop Proceedings., CEUR-WS.org (2013) 168–180
11. Giordano, L., Gliozzi, V., Olivetti, N., Pozzato, G.L.: Minimal model semantics and rational closure in description logics. In: Informal Proceedings of Description Logics 2013. Volume 1014 of CEUR Workshop Proceedings., CEUR-WS.org (2013) 168–180
12. Pozzato, G.: Typicalities and probabilities of exceptions in nonmotonic Description Logics. International Journal of Approximate Reasoning **107** (2019) 81–100
13. Giordano, L., Gliozzi, V., Olivetti, N., Pozzato, G.L.: A NonMonotonic Description Logic for Reasoning About Typicality. Artificial Intelligence **195** (2013) 165 – 202
14. Pozzato, G.L.: Reasoning about surprising scenarios in description logics of typicality. In: Advances in Artificial Intelligence: AI*IA 2016. Volume 10037 of LNAI. (2016) 1–15
15. Riguzzi, F., Bellodi, E., Lamma, E., Zese, R.: Probabilistic description logics under the distribution semantics. Semantic Web **6**(5) (2015) 477–501
16. Giordano, L., Gliozzi, V., Olivetti, N., Pozzato, G.L.: Rational closure in $\mathcal{SHIQ}$. In: Proc. of Description Logics 2014. Volume 1193 of CEUR Workshop Proceedings., CEUR-WS.org (2014) 543–555
17. Lieto, A., Pozzato, G.L.: A description logic of typicality for conceptual combination. In: Proc. of ISMIS 2018. Volume 11177 of LNAI., Springer (2018) 189–199
18. Alberti, M., Bellodi, E., Cota, G., Riguzzi, F., Zese, R.: cplint on SWISH: probabilistic logical inference with a web browser. Intelligenza Artificiale **11**(1) (2017) 47–64
19. Bellodi, E., Lamma, E., Riguzzi, F., Zese, R., Cota, G.: A web system for reasoning with probabilistic OWL. Journal of Software: Practice and Experience **47**(1) (2017) 125–142